REMARKS

Claims 1-24 are currently pending in the subject application and are presently under consideration. Favorable consideration of the subject patent application is respectfully requested in view of the comments herein.

I.   Rejection of Claim 23 Under 35 U.S.C. §101

Claim 23 stands rejected under 35 U.S.C. §101 because the claimed invention lacks patentable utility. Withdrawal of this rejection is requested for at least the following reasons. Claim 23 produces a useful, concrete and tangible result, and further, the subject claim pertains to the transmission of software code between two or more computer processes.

> Because the claimed process applies the Boolean principle [abstract idea] *to produce a useful, concrete, tangible result* ... on its face the claimed process comfortably falls within the scope of §101. *AT&T Corp. v. Excel Communications, Inc.*, 172 F.3d 1352, 1358. (Fed. Cir. 1999) (Emphasis added); *See State Street Bank & Trust Co. v. Signature Fin. Group, Inc.*, 149 F.3d 1368, 1373, 47 USPQ2d 1596, 1601 (Fed.Cir.1998). The inquiry into patentability requires an examination of the contested claims to see if the claimed subject matter, as a whole, is a disembodied mathematical concept representing nothing more than a "law of nature" or an "abstract idea," or if the mathematical concept has been *reduced to some practical application rendering it "useful."* *AT&T* at 1357 *citing In re Alappat*, 33 F.3d 1526, 31 1544, 31 U.S.P.Q.2D (BNA) 1545, 1557 (Fed. Cir. 1994) (emphasis added).

The Examiner contends: "[a] communications signal transmitting between two computers a data packet is not tangible." Applicants' representative disagrees and submits that the Examiner is misconstruing the requirements necessary to fulfill the conditions for patentability under 35 U.S.C. §101. According to *AT&T Corp. v. Excel Communications, Inc.*, 172 F.3d 1352 (Fed. Cir. 1999), the standard set forth by the Federal Circuit for determining whether claims are directed towards statutory subject matter is whether the claims as a whole can be applied in a practical application to *produce a useful, concrete and tangible result.* It is the result of the claims as applied in a practical application that is germane to the determination of whether the claims are

- 2 -

directed towards statutory subject matter, not whether the underlying means by which the result is effectuated that should be tangible, as the Examiner intimates. It is believed therefore that the subject claim clearly satisfies this legal standard. In particular, independent claim 23 recites: *a data packet transmitted as a communication signal between at least two computer processes, comprising: a configurable module having: one or more configurable data elements, wherein one or more default values for the one or more configurable data elements are available; one or more non-configurable data elements describing the one or more configurable data elements; and one or more transformation instructions that facilitate configuring the one or more configurable data elements, wherein the instructions are employed to facilitate installation of the one or more configurable data elements into a target data set residing in at least one of the at least two computer processes.* Thus, claim 23 elicits a series of independent acts that culminates in a useful, concrete and tangible result – the installation of the one or more configurable data elements into a target data set residing in at least one of the at least two computer processes.

Additionally, the Court of Appeals for the Federal Circuit stated in *Eolas Techs., Inc. v. Microsoft Corp.*, 399 F.3d 1325 (Fed. Cir. 2005):

> Title 35, section 101, explains that an invention includes "any new and useful process, machine, manufacture or composition of matter." ... Without question, *software code alone qualifies as an invention eligible for patenting under these categories*, at least as processes. *Id.* at 1338 (emphasis added).

The subject claim clearly pertains to software code comprising a configuration module, one or more configurable data elements, one or more non-configurable data elements, and one or more transformation instructions encapsulated in a data packet transmitted from one computer process to another computer process, so that the transformation instructions therein can be employed to facilitate installation of the one or more configurable data elements into a target data set residing in at least one of the at least two computer processes. The fact that (i) the data packet that encases the software code during its transmission between two processes, or (ii) the data packet is transmitted as a communication signal between two processes is irrelevant to the fact that it is software

- 3 -

code that is contained therein and is being transmitted through utilization of a communication signal. It is submitted that all that is relevant is the fact that software code is being transmitted within the data packet, and that the software code so transmitted produces a useful, concrete and tangible result. Accordingly, withdrawal of this rejection is requested.

II.   Rejection of Claims 1-24 Under 35 U.S.C. §102(b)

Claims 1-24 stand rejected under 35 U.S.C. §102(b) as being anticipated by Microsoft's Visual C++ version 5.0 as documented in the text book, "Beginning Visual C++ 5", by Ivor Horton, published March 19, 1997 (hereinafter "Horton"). This rejection should be withdrawn for at least the following reasons. Horton does not disclose or suggest each and every limitation set forth in the subject claims.

> A *single prior art reference* anticipates a patent claim only if it *expressly or inherently describes each and every limitation set forth in the patent claim. Trintec Industries, Inc. v. Top-U.S.A. Corp.*, 295 F.3d 1292, 63 USPQ2d 1597 (Fed. Cir. 2002); *See Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). The *identical invention must be shown in as complete detail as is contained in the ... claim. Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989) (emphasis added).

Applicants' claimed invention relates to a system and method for configuring software components that are incorporated into computer programs. In particular, the subject invention relates to creating, describing and configuring software components (that are data sets) that are to be incorporated into software programs (target data sets) such that the components are self-describing in relation to configuration possibilities for the software components. To this end, independent claims 1, 9, 13, 17, 22, 23 and 24 recite similar claim limitations, namely: ... *installation of the one or more configurable data elements into at least one target data set*. Horton does not disclose or suggest these novel aspects of the invention as claimed.

The Examiner contends that Horton provides substantiation for the rejection of the subject claims. Applicants' representative disagrees. Horton discloses the creation

- 4 -

and utilization of dynamic link libraries, and specifically that a dynamic link library is a file containing a collection of modules that can be used by any number of different programs. (*See* page 717, lines 1-2). The cited document further discloses that no code from the dynamic link library is ever included in the executable module of any of the programs, such that when one of the programs is executed, the program is loaded into memory, and if the dynamic link library that the program utilizes is not already present in memory, it too *is loaded separately into memory*. (*See* page 718, lines 2-5). Horton also specifically notes that when a program calls a function in a dynamic link library, the operating system will automatically *load the dynamic link library into memory*, and as a · consequence, since the operating system recognizes that the library is already in memory, any program that is subsequently loaded into memory that utilizes the same dynamic link library can use any of the capabilities provided by the same copy of the dynamic link library; all that is required is that links between the program and the dynamic link library be established. The cited document further notes that when a dynamic link library is no longer used by any executing program, the operating system will automatically *delete the dynamic link library from memory*. (*See* page 718, lines 8-15). From the foregoing it is apparent that the dynamic link library as disclosed in Horton is a separate, distinct and transient entity that is loaded into memory to operate in conjunction with a program that is also loaded into memory, and that once the dynamic link library has fulfilled its purpose it is deleted from the memory. The invention as claimed in contrast, installs and/or loads configurable data elements *into* a target data set through the use of transformational instructions applied to configurable data elements to transform the target data set with the configurable elements obtained from a transformation component, *e.g.*, when the configurable data elements are installed and/or loaded into a target data set there is a transformation in that the target data set is no longer the entity it was prior to utilization of the claimed invention; the target data set now has installed therein configurable data elements.

Additionally, it is evident that the program and the dynamic link library that are loaded into memory always remain distinct entities. In particular, the definitional section provided by the Examiner, *viz.* Term's in the Art, reinforces the perception that the dynamic link library and the program always remain separate and distinct entities in that

definition (A) provides that dynamic link libraries are "executable routines [that are] stored separately as files ... [and are] loaded when needed by a program", and definition (B) states that a dynamic link library is "a file containing executable code and data bound to a program at load time, or run time, rather than during linking." The import of these two definitions is that they are an indication that dynamic link libraries are never installed within/into the target/invoking program, but rather that the invoked dynamic link library is merely linked to the invoking program at some point when the functionality of the dynamic link library is required. Thus, it is submitted that the dynamic link library always remains distinct from the target/invoking program, and that the dynamic link library is never installed into the target/invoking program.

In addition, the Examiner in the Response to Arguments section of the Final Office Action dated June 28, 2005, states "the end result of the merge of the content of the DLL (software AND/OR data) is the content of the DLL is merged into a target data set (the software component(s)) program which the content is being modified." *Id.* at page 8. Horton does not teach or suggest that there is a merger of the dynamic link library into the target data set/program, and neither do the definitions upon which the Examiner places reliance. All that Horton (and the definitions provided by the Examiner) discloses is that dynamic link libraries are loaded into memory when needed by an invoking program, and that links are created between the invoking program and the dynamic link library invoked. No where in Horton is there a discussion of merging the dynamic link library into the calling program let alone installing one or more configurable data elements into at least one target data set as recited in the subject claims.

Furthermore, the Examiner is reminded that the standard by which anticipation is to be measured is *strict identity* between the cited document and the invention as claimed, not mere equivalence or similarity. *See, Richardson* at 9 USPQ2d 1913, 1920. This means that in order to establish anticipation under 35 U.S.C. §102, *a single document* must not only expressly or inherently describe *each and every* limitation set forth in the patent claim, but also *the identical invention must be shown in as complete detail as contained in the claim.* Horton alone, despite the supposed "Term's in the Art" supplied by the Examiner, does not provide the required identity to substantiate this rejection under 35 U.S.C. §102. In particular, the cited document does not provide for the

- 6 -

09/770,705   MS155721.02/MSFTP139USA

installation of configurable data elements into a target data set thereby rendering a single indivisible entity – a software program that also includes configurable date elements installed therein.

In view of at least the foregoing, it is respectfully requested that the rejection of independent claims 1, 9, 13, 17, 22, 23 and 24 (and claims that depend there from) should be withdrawn.

CONCLUSION

The present application is believed to be in condition for allowance in view of the above comments. A prompt action to such end is earnestly solicited.

In the event any fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063[MSFTP139USA].

Should the Examiner believe a telephone interview would be helpful to expedite favorable prosecution, the Examiner is invited to contact applicants' undersigned representative at the telephone number below.

Respectfully submitted,

AMIN & TUROCY, LLP

Himanshu S. Amin
Reg. No. 40,894

AMIN & TUROCY, LLP
24TH Floor, National City Center
1900 E. 9TH Street
Cleveland, Ohio 44114
Telephone (216) 696-8730
Facsimile (216) 696-8731

- 7 -